

In the Claims:

1. (Previously Presented) A method of transmitting accounting records in an accounting system that produces information pertaining to network traffic flow comprising:
- collecting data from a network device by a data collector associated with the network device and producing accounting records from the data;
 - transmitting the accounting records to first and second flow aggregation processes, with transmitting further comprising for each flow aggregation process:
 - storing in the data collector the accounting records;
 - transmitting the accounting records to the flow aggregation process; and
 - awaiting an acknowledgment signal from the flow aggregation process that the flow aggregation process received the accounting records before discarding the accounting records sent to the flow aggregation process.
2. (Currently Amended) A system, including an accounting module, comprising:
- a plurality of data collectors disposed in a network, and which collect data pertaining to operation of the network from network devices;
 - a first flow aggregation process, connected to the plurality of data collectors, wherein the data collectors send the collected data to the first flow aggregation process; and
 - wherein the data collectors dispose of the collected data only after receiving an acknowledgement that the data has been received with the first flow aggregation process processing the data to generate aggregated records; and
 - a second flow aggregation process, connected to the data collectors, wherein the data collectors send the collected data to the second flow aggregation process, and dispose of the collected data only after receiving an acknowledgement that the data has been received, with the second flow aggregation process ~~processes~~ processing the data to generate aggregated records.
3. (Previously Presented) The method of claim 1 wherein if the data collector determines that the flow aggregation process is not operating, the method further comprises:
- continuing to collect and store accounting records from the network device for future transmission to that flow aggregation process.

OK
TOE/121
12/10.8.4

BEST AVAILABLE COPY

4. (Previously Presented) The method of claim 1 wherein if the data collector does not receive an acknowledgement signal in response to transmitting the records to the flow aggregation process, the method further comprises:
- determining an error relating to the first flow aggregation process; and
 - causing aggregate reports from the second flow aggregation process to be sent to the accounting module in place of the aggregate reports from the first flow aggregation process.
5. (Previously Presented) The method of claim 1 wherein the data collector produces network accounting records (NARs) from collected data.
6. (Previously Presented) The method of claim 5 wherein if the transfer is successful, the method further comprises:
- removing from a local store of the data collector the locally stored copies of the transferred NARs.
7. (Currently Amended) The method of claim 5 wherein store and forward capabilities of the flow data collector provide fault tolerance at this accounting process level to ensure reliable data transfer.
8. (Currently Amended) The method of claim 5 wherein the flow data collector only transfers NARs when the data collector has determined that the flow aggregation process is available, and the data collector considers the NAR transfer successful upon receipt of an acknowledgement from the flow aggregation process.
9. (Currently Amended) The method of claim 1 further comprises:
- determining an error relating to the first flow aggregation process, and causing aggregate reports from the second flow aggregation process to be sent to the accounting module in place of the aggregate reports from the first flow aggregation process.

BEST AVAILABLE COPY

10. (Currently Amended) The system of claim 2 wherein the first and the second flow aggregation ~~process~~ ~~processes~~ ~~summarizes~~ ~~summarize~~ related information from the received NARs across network device.
11. (Currently Amended) The system of claim 2 further comprising:
an error detection process that detects an error relating to the first flow aggregation process[[,]] to cause the aggregate reports from the second flow aggregation process to be sent to the accounting module in place of the aggregate reports from the first flow aggregation process.
12. (Previously Presented) The system of claim 11 wherein the data collector further comprises:
logic to determine that the flow aggregation process is not operating to cause the data collector to collect and store accounting records from the network device for future transmission to the flow aggregation process.
13. (Currently Amended) The system of claim 11 wherein the data collector[[,]] further comprises:
logic to determine an error relating to the first flow aggregation process; and to cause aggregate reports from the second flow aggregation process to be sent to the accounting module in place of the aggregate reports from the first flow aggregation process.
14. (Previously Presented) The system of claim 11 wherein the data collector produces network accounting records (NARs) from collected data.
15. (Currently Amended) The system of claim 14 wherein the data ~~collectors~~ collector further ~~comprise~~ comprises:
a local store that locally ~~stored~~ stores copies of the transferred NARs.
16. (Previously Presented) The system of claim 11 wherein the data collectors provide store and forward capabilities to provide fault tolerance to the accounting module to ensure reliable data transfer.

17. (Previously Presented) The system of claim 11 wherein the data collector only transfers NARs when the data collector has determined that the flow aggregation process is available, and the data collector considers the NAR transfer successful upon receipt of an acknowledgement from the flow aggregation process.

18. (Currently Amended) A computer program product residing on a computer readable medium for transmitting accounting records in an accounting system comprising instructions to cause a computer to:

collect data associated with a network device and produce accounting records from the data;

transmit the accounting records to first and second aggregation flow processes to produce aggregate reports of the accounting records;

store the accounting records in a local storage; and

await an acknowledgement signal from the first flow aggregation process that the flow aggregation process received the accounting records before discarding the accounting records sent to the first flow aggregation process; and

determine an error relating to the first flow aggregation process to cause the aggregate reports from the second flow aggregation process to be sent to the accounting module in place of the aggregate reports from the first flow aggregation process.

19. (Currently Amended) The computer program product of claim 18 further comprising instructions to cause a computer to:

determine that one of the flow aggregation processes is not operating; and

cause the a data collector to continue to collect and store records from the network device for future transmission to that flow aggregation process.

20. (Previously Presented) The computer program product of claim 19 wherein the data collector produces network accounting records (NARs) from collected data.

21. (Previously Presented) The computer program product of claim 19 further comprises instructions to:

remove the locally stored copies of the transferred NARs from a local store of the data collector if the transfer is successful.

22. (Currently Amended) The computer program product of claim 19 wherein store and forward capabilities of the data collector provide fault tolerance at the accounting process to ensure reliable data transfer.

23. (Currently Amended) The computer program product of claim 19 further comprises instructions to:

determine that the aggregation process is available before the data collector transfer NARs, and the data collector considers the NAR transfer is to be successful upon receipt of an acknowledgement from the aggregation process.

BEST AVAILABLE COPY